KEYEN Communication Driver

Driver for Serial communication with
Keyence devices (KV and KV-P Series)

Contents

Introduction

The KEYEN driver enables communication between the Studio system and devices using the Keyence Protocol over Serial communication, according to the specifications discussed in this document.

This document will help you to select, configure and execute the KEYEN driver, and it is organized as follows:

- **Introduction**: This section, which provides an overview of the document.
- **General Information**: Identifies all of the hardware and software components required to implement communication between the Studio system and the target device.
- **Selecting the Driver**: Explains how to select the KEYEN driver in the Studio system.
- **Configuring the Device**: Describes how the target device must be configured to receive communication from the KEYEN driver.
- **Configuring the Driver**: Explains how to configure the KEYEN driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver**: Explains how to execute the KEYEN driver during application runtime.
- **Troubleshooting**: Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Sample Application**: Explains how to use a sample application to test the KEYEN driver configuration
- **Revision History**: Provides a log of all changes made to the driver and this documentation.

Notes:

- This document assumes that you have read the “Development Environment” chapter in Studio’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows 2000/XP/Vista environment. If you are not familiar with Windows, then we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter identifies all of the hardware and software components required to implement communication between the KEYEN driver in Studio and a target Keyence device (KV and KV-P Series) over Serial communication.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Testing

Device Specifications

To establish communication, your target device must meet the following specifications:

- **Manufacturer**: KEYENCE
- **Compatible Equipment**:
  - • KV Series
  - • KV-P Series
- **Device Runtime Software**: None

For a description of the device(s) used to test driver conformance, see “Conformance Testing”.

Network Specifications

To establish communication, your device network must meet the following specifications:

- **Device Communication Port**: Communication Port A (Serial)
- **Physical Protocol**: RS232
- **Logic Protocol**: Proprietary
- **Device Runtime Software**: None
- **Specific PC Board**: None
- **Cable:** OP-26486 (cable) and OP-26487 (9-Pin Sub D Connector)

Driver Characteristics

The KEYEN driver package consists of the following files, which are automatically installed in the **`/DRV`** subdirectory of Studio:

- **`KEYEN.INI:`** Internal driver file. *You must not modify this file*.
- **`KEYEN.MSG:`** Internal driver file containing error messages for each error code. *You must not modify this file*.
- **`KEYEN.PDF:`** This document, which provides detailed information about the KEYEN driver.
- **`KEYEN.DLL:`** Compiled driver.

> **Note:**
> You must use Adobe Acrobat® Reader™ to view the **`KEYEN.PDF`** document. You can install Acrobat Reader from the Studio installation CD, or you can download it from Adobe’s Web site.

You can use the KEYEN driver on the following operating systems:

- Windows 2000/XP/Vista
- Windows CE

For a description of the operating systems used to test driver conformance, see “Conformance Testing” below.

The KEYEN driver supports the following registers:

Type		Write	Read
Counter (C)	Current (CUR)	•	•
	Preset (PRE)	•	•
	Contact (CNT)	•	•
High-speed counter (CTH)	Current (CUR)	•	•
	Preset (PRE)	–	•
	Contact (CNT)	–	•
High-speed counter comparator (CTC)	Current (CUR)	–	•
	Preset (PRE)	•	•
	Contact (CNT)	•	•
Timer (T)	Current (CUR)	•	•
	Preset (PRE)	•	•
	Contact (CNT)	•	•
Digital trimmer (AT)	Trimmer 0 (0)	–	•
	Trimmer 1 (1)	–	•
Relay contact (R)	Contact	•	•
Data memory (DM)	Value	•	•
Temporary data memory (TM)	Value	•	•

Note:
- To know about registers length consult your User’s manual, it depends on each device type.

Conformance Testing

The following hardware/software was used for conformance testing:

- **Driver Configuration**:
  - **Baud Rate**: 9600
  - **Data Bits**: 8
  - **Stop Bits**: 1
  - **Parity**: Even

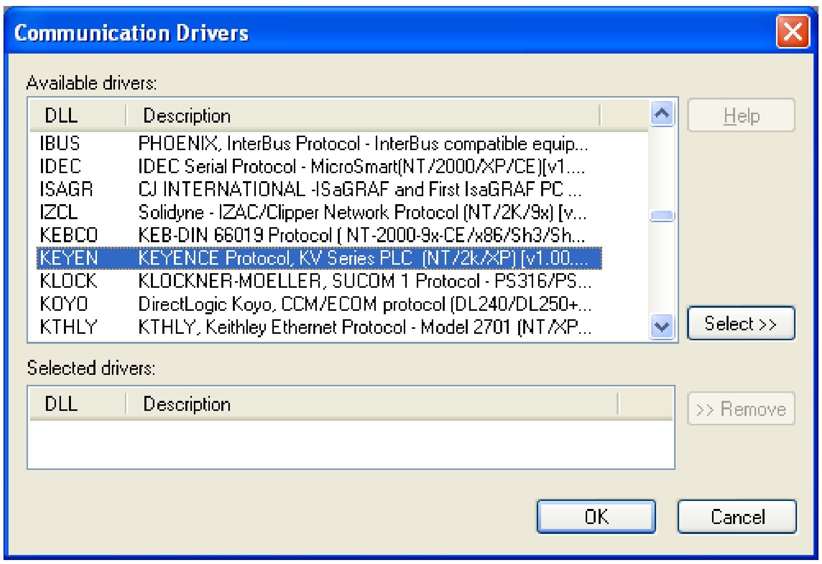
- **Cable**: Use specifications described in the “**Error! Reference source not found.**” section above

Driver Version	Studio Version	Operating System (development)	Operating System (runtime)	Equipment
1.00	6.1 + SP4	Win XP + SP2	▪ Win XP+SP2	▪ KV-16AR ▪ KV-10T ▪ KV-16DT ▪ KV-P16RL

Selecting the Driver

When you install Studio, all of the communication drivers are automatically installed in the **\DRV** subdirectory but they remain dormant until manually selected for specific applications. To select the KEYEN driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.
2. Select the **KEYEN** driver from the *Available Drivers* list, and then click the **Select** button.



Communication Drivers Dialog

3. When the **KEYEN** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

> **Note:**
> It is not necessary to install any other software on your computer to enable communication between the host and the device. However, to download the custom program to your device, you must install the KEYENCE programmer software (Ladder Builder for example). Consult your KEYENCE programmer software documentation for installation instructions.

> **Caution:**
> For safety reasons, you must take special precautions when installing any physical hardware. Please consult the manufacturer's documentation for specific instructions.

Configuring the Device

Once the selected driver and the target device are both properly configured, it is not necessary to install any other software on your computer to enable communication between the host and the device. All runtime communication is handled within your Studio application project. However, programming the device itself — that is, developing control logic and downloading it to the device — still requires using the device's own programming tool.

Configuring the Driver

Once you have selected the KEYEN driver in Studio, you must properly configure it to communicate with your target device. First, you must set the driver's communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio application with the appropriate addresses (registers) on the device.

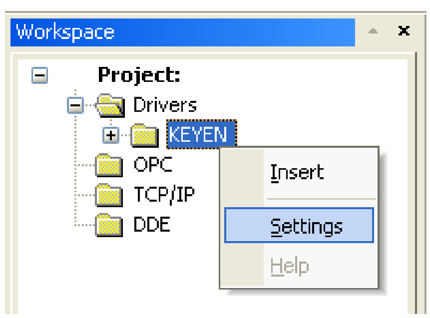
Configuring the Communication Settings

The communication settings are described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only KEYEN driver-specific settings and procedures will be discussed here. To configure the communication settings for the KEYEN driver:

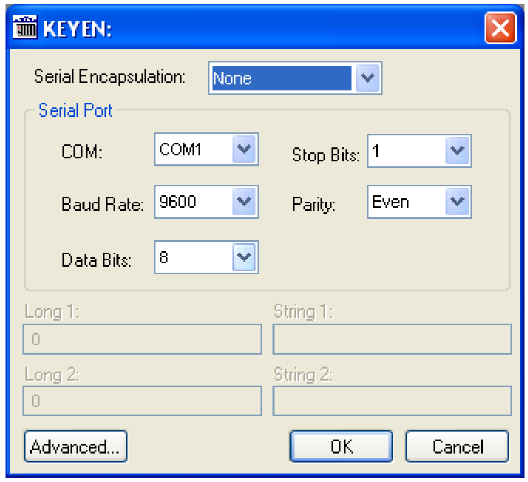
1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The KEYEN driver is listed here as a subfolder.

2. Right-click on the *KEYEN* subfolder and then select the **Settings** option from the pop-up menu:



Select Settings from the Pop-Up Menu

The *KEYEN: Communication Settings* dialog is displayed:



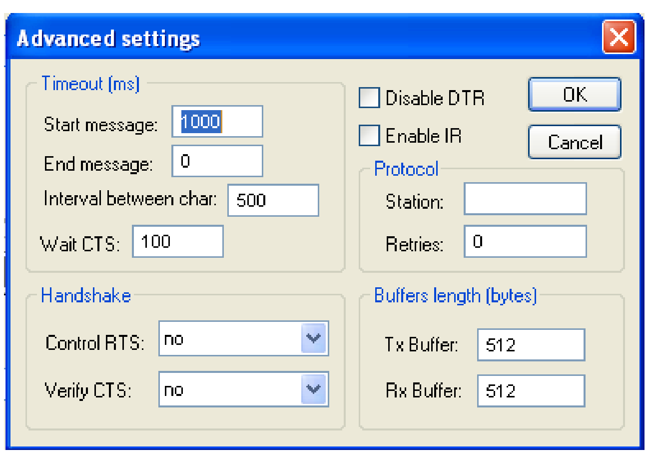
KEYEN: Communication Settings Dialog

3. In the *Communication Settings* dialog, configure the driver settings to enable communication with your target device. To ensure error-free communication, the driver settings must *exactly match* the corresponding settings on the device. Please consult the manufacturer's documentation for instructions how to configure the device and for complete descriptions of the settings.

   Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

> **Caution:**
> For safety reasons, you **must** take special precautions when connecting and configuring new equipment. Please consult the manufacturer's documentation for specific instructions.

4. If you are using a Data Communication Equipment (DCE) converter (e.g., 232/485) between your PC and your target device, then you must also adjust the **Control RTS** (Request to Send) setting to account for the converter. In the *Communication Settings* dialog, click the **Advanced** button to open the *Advanced Settings* dialog:



Advanced Settings Dialog

When the dialog is displayed, configure the **Control RTS** setting using the following information:

Setting	Default	Values	Description
Control RTS	no	no	Do not set the RTS (Request to Send) handshake signal. IMPORTANT: If you are using Windows 95/98 or Windows CE with the correct RS232/RS485 adapter (i.e. without RTS control), then you must select this option.
		yes	Set the RTS (Request to Send) handshake signal before communication. IMPORTANT: If you are using Windows NT and the Cutler-Hammer RS232/RS485 adapter, then you must select this option.
		yes+echo	Set the RTS (Request to Send) handshake signal before communication, and echo the signal received from the target device.

➲ **Cautions**:
If you incorrectly configure the **Control RTS** setting, then runtime communication will fail and the driver will generate a –15 error. See “Troubleshooting” for more information.

You do not need to change any other advanced settings at this time. You can consult the Studio *Technical Reference Manual* later for more information about configuring these settings.

5. Click **OK** to close the *Advanced Settings* dialog, and then click **OK** to close the *Communication Settings* dialog.

Configuring the Driver Worksheets

Each selected driver includes a one or more Standard Driver Worksheets. Standard Driver Worksheets can be inserted to define additional tag/register associations that are triggered by specific application behaviors.

The configuration of these worksheets is described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

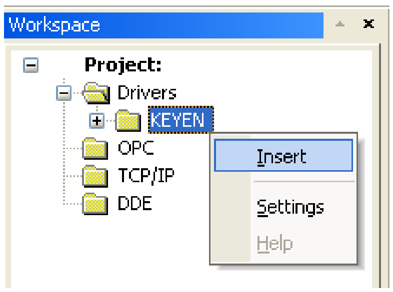
For the purposes of this document, only KEYEN driver-specific parameters and procedures are discussed here.

STANDARD DRIVER WORKSHEET

When you select the KEYEN driver, you may insert Standard Driver Worksheets to define tag/register associations that are triggered by specific application behaviors. Doing this will optimize communication and improve system performance by ensuring that tags/registers are scanned only when necessary – that is, only when the application is performing an action that requires reading or writing to those specific tags/registers.

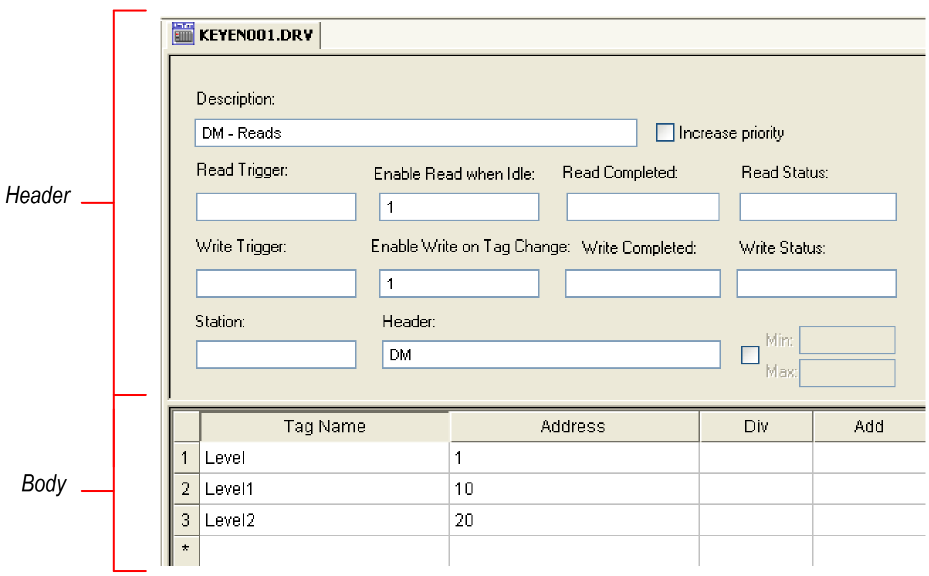
To insert a new Standard Driver Worksheet:

1. In the *Comm* tab, open the *Drivers* folder and locate the *KEYEN* subfolder.
2. Right-click on the *KEYEN* subfolder, and then select **Insert** from the pop-up menu:



Inserting a New Worksheet

A new KEYEN driver worksheet is inserted into the *KEYEN* subfolder, and the worksheet is opened for configuration:



KEYEN Driver Worksheet

Most of the fields on this worksheet are standard for all drivers; see the "Communication" chapter of the *Technical Reference Manual* for more information on configuring these fields. However, the **Station**, **Header**, and **Address** fields use syntax that is specific to the KEYEN driver.

3. Configure the **Station** and **Header** fields as follows:

   - **Station** field: Not used for this driver

   - **Header** field: Specify the address of the first register of a block of registers on the target device. The addresses declared in the *Body* of the worksheet are simply offsets of this **Header** address. When Read/Write operations are executed for the entire worksheet (see **Read Trigger** and **Write Trigger** above), it scans the entire block of registers from the first address to the last.

     The **Header** field uses the following syntax:

     <Type>

     Example — **C; CTH; DM**

     Where:

     <Type> is the register type. (C, CTH, CTC, T, AT, R, DM and TM)

     After you edit the **Header** field, Studio checks the syntax to determine if it is valid. If the syntax is invalid, then Studio automatically inserts a default value of **C**

     You can also specify an indirect tag (e.g. **{header}**), but the tag that is referenced must follow the same syntax and contain a valid value.

> **Caution:**
> - You cannot leave the **Header** fields blank; you must specify some values.

> **Note:**
> - See the supported registers in the Driver Characteristics session.

- **Address** field: For each table row (i.e., each tag/register association), configure the **Address** field using the following syntax…

  <Address>.[SubType]

  Where:

  – ***<Address>:*** is the specific address of the register on the device.

  – ***[SubType]:*** is an optional parameter that must be used only with C, CTH, CTC and T headers.

  The options for this parameter are:

  CUR (Current);

  PRE (Preset) and

  CNT (Contact).

For examples of how register addresses are composed using the values in the **Header** and **Address** fields, see the following table:

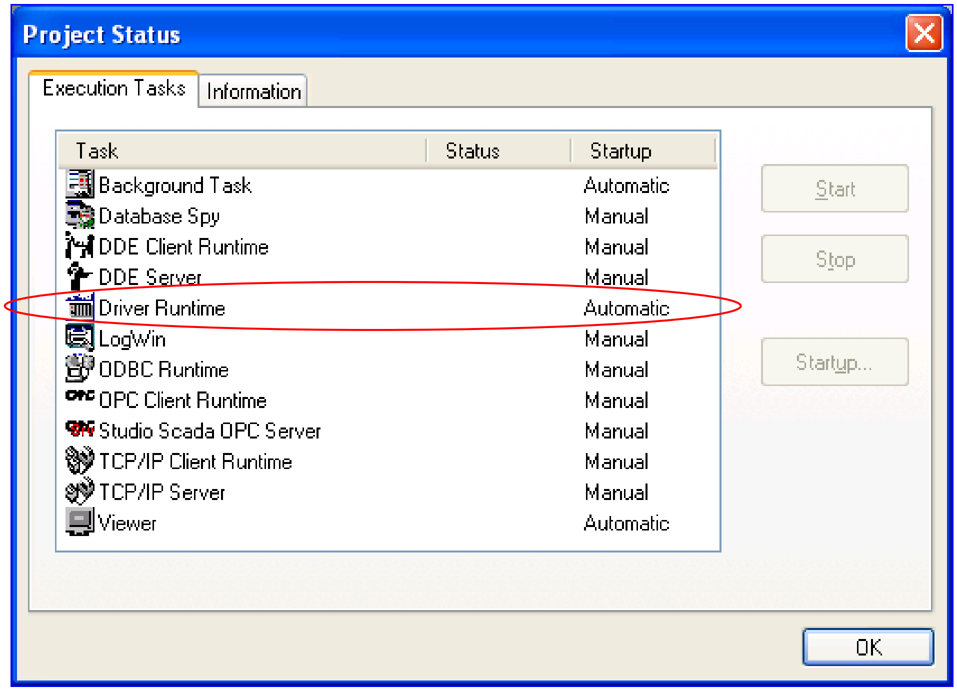
Header	Address Reference	SubType	Address Field (examples)
C	0 - 999	CUR (Current Value)	1.CUR
		PRE (Preset Value)	50.PRE
		CNT (Contact Value)	212.CNT
CTH	0 - 9	CUR (Current Value)	5.CUR
		PRE (Preset Value)	3.PRE
		CNT (Contact Value)	4.CNT
CTC	0 – 9	CUR (Current Value)	2.CUR
		PRE (Preset Value)	6.PRE
		CNT (Contact Value)	8.CNT
T	0 - 999	CUR (Current Value)	5.CUR
		PRE (Preset Value)	98.PRE
		CNT (Contact Value)	102.CNT
AT	0 (Trimmer 0)	-	0
	1 (Trimmer 1)		1
R	0 - 65535	-	100
DM	0-9999	-	30
TM	0-99	-	7

Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application's runtime execution settings by checking the *Project Status* dialog.

To verify that the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.

   - If the setting is correct, then proceed to step 3 below.
   - If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task's *Startup* mode to **Automatic**.

3. Click **OK** to close the *Project Status* dialog.

4. Start the application to run the driver.

Troubleshooting

If the KEYEN driver fails to communicate with the target device, then the database tag(s) that you configured for the **Read Status** or **Write Status** fields of the Driver Sheet will receive an error code. Use this error code and the following table to identify what kind of failure occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	▪ Communication without problems	▪ None required
1	INVALID PROTOCOL	▪ The received message is invalid.	▪ Check the worksheet configuration
2	INVALID COMMAND	▪ Driver Worksheet is configured wrong.	▪ Check the worksheet configuration
3	ERROR START COMMUNICATION	▪ Error starting the communication.	▪ Check cable wiring ▪ Check the PLC state – it must be RUN. ▪ Check the configuration. See *Studio Technical Reference Manual* for information about valid RTS/CTS configurations
4	ERROR END COMMUNICATION	▪ Error finishing the communication.	▪ Check cable wiring ▪ Check the PLC state – it must be RUN. ▪ Check the configuration. See *Studio Technical Reference Manual* for information about valid RTS/CTS configurations
-15	Timeout Start Message	▪ Disconnected cables ▪ PLC is turned off, in stop mode, or in error mode ▪ Wrong RTS/CTS control settings	▪ Check cable wiring. ▪ Check the PLC state – it must be RUN. ▪ Check the configuration. See *Studio Technical Reference Manual* for information about valid RTS/CTS configurations.
-17	Timeout between rx char	▪ PLC in stop mode or in error mode ▪ Wrong parity ▪ Wrong RTS/CTS configuration settings	▪ Check cable wiring. ▪ Check the PLC state – it must be RUN. ▪ Check the configuration. ▪ Check the configuration. See *Studio Technical Reference Manual* for information about valid RTS/CTS configurations.

⇨ **Tip:**

You can monitor communication status by establishing an event log in Studio's *Output* window (*LogWin* module). To establish a log for **Field Read Commands**, **Field Write Commands** and **Serial Communication,** right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the *LogWin* module (**Tools** → **LogWin**) to establish an event log on a remote unit that runs Windows CE. The log is saved on the unit in the `celog.txt` file, which can be downloaded later.

If you are unable to establish communication between Studio and the target device, then try instead to establish communication using the device's own programming software. Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

To test communication between Studio and the device, we recommend using the sample application provided rather than your new application.

If you must contact us for technical support, please have the following information available:

- **Operating System** (type and version): To find this information, select **Tools → System Information.**
- **Project Information**: To find this information, select **Project → Status.**
- **Driver Version** and **Communication Log**: Displays in the Studio *Output* window when the driver is running.
- **Device Model** and **Boards**: Consult the hardware manufacturer's documentation for this information.

Sample Application

There was not an official sample application available for this driver by the timer that this document was written.

Revision History

Doc. Revision	Driver Version	Author	Date	Description of changes
A	1.00	Eric Vigiani	Dec/04/2003	Initial Version
B	1.00	Eric Vigiani	Mar/12/2008	Doc. revision
B	1.01	Lourenço Teodoro	Dec/16/2008	Updated driver version, no change in the contents.
C	1.1	Eduardo Castro	Sep/16/2014	Updated driver version, no change in the contents.